Parsimonious Relevance Models

Edgar Meij
emeij@science.uva.nl

Wouter Weerkamp
weerkamp@science.uva.nl

Krisztian Balog
kbalog@science.uva.nl

Maarten de Rijke
mdr@science.uva.nl

ISLA, University of Amsterdam
Kruislaan 403, 1098 SJ Amsterdam

ABSTRACT

We describe a method for applying parsimonious language models to re-estimate the term probabilities assigned by relevance models. We apply our method to six topic sets from test collections in five different genres. Our parsimonious relevance models (i) improve retrieval effectiveness in terms of MAP on all collections, (ii) significantly outperform their non-parsimonious counterparts on most measures, and (iii) have a precision enhancing effect, unlike other blind relevance feedback methods.

Categories and Subject Descriptors

H.3 [**Information Storage and Retrieval**]: H.3.3 Information Search and Retrieval

General Terms

Algorithms, Theory, Experimentation, Measurement

Keywords

Parsimonious Models, Language Models, Relevance Feedback

1. INTRODUCTION

Relevance feedback is often applied to better capture a user's information need [1, 5, 12]. Automatically reformulating queries (or *blind* relevance feedback) entails looking at the terms in some set of (pseudo-)relevant documents and selecting the most informative ones with respect to the set or the collection. These terms may then be reweighed based on information pertinent to the query or the documents and—in a language modeling setting—be used to estimate a query model, $P(t|\theta_Q)$, i.e., a distribution over terms t for a given query Q [7, 13].

Not all of the terms obtained using blind relevance feedback are equally informative given the query, even after reweighing. Some may be common terms, whilst others may describe the general domain of interest. We hypothesize that refining the results of blind relevance feedback, using a technique called parsimonious language modeling [3], will improve retrieval effectiveness. Hiemstra et al. [3] already provide a mechanism for incorporating (parsimonious) blind relevance feedback, by viewing it as a three component mixture model of document, set of feedback documents, and collection. Our approach is more straightforward, since it considers each feedback document separately and, hence, does not require the additional mixture model parameter. To create parsimonious language models we use an EM algorithm to update the maximum-likelihood (ML) estimates. Zhai and Lafferty [13] already proposed an approach which uses a similar EM algorithm; it differs, however, in the way the set of feedback documents is handled. Whereas we parsimonize each individual document, they apply their EM algorithm to the entire set of feedback documents.

To verify our hypothesis, we use a specific instance of blind relevance feedback, namely relevance modeling (RM) [5]. We choose this particular method because it has been shown to achieve state-of-the-art retrieval performance. Relevance modeling assumes that the query and the set of documents are samples from an underlying term distribution—the relevance model. Lavrenko and Croft [5] formulate two ways of approaching the estimation of the parameters of this model. We build upon their work and compare the results of our proposed parsimonious relevance models with RMs as well as with a query-likelihood baseline. To measure the effects in different contexts, we employ five test collections taken from the TREC-7, TREC Robust, Genomics, Blog, and Enterprise tracks and show that our proposed model improves performance in terms of mean average precision on all the topic sets over both a query-likelihood baseline as well as a run based on relevance models. Moreover, although blind relevance feedback is mainly a recall enhancing technique [9], we observe that parsimonious relevance models (unlike their non-parsimonized counterparts) can also improve early precision and reciprocal rank of the first relevant result.

2. PARSIMONIOUS RELEVANCE MODELS

Relevance models use a set of (pseudo-)relevant documents $\mathcal{D}_Q$ to estimate a query model $P(t|\theta_Q)$. We use method 2, as proposed by Lavrenko and Croft [5]:

$$P(t|\hat{\theta}_Q) \propto P(t) \cdot \prod_{i=1}^{k} \sum_{D_i \in \mathcal{D}_Q} P(q_i|D_i) \cdot P(D_i|t), \qquad (1)$$

where $q_1, \ldots, q_k$ are the query terms, $P(D_i|t) = \frac{P(t|D_i) \cdot P(D_i)}{P(t)}$, and

$$P(t|D_i) \quad = \quad 0.5 \cdot \frac{c(t; D_i)}{\sum_{t'} c(t'; D_i)} + 0.5 \cdot P(t|C), \qquad (2)$$

where $c(t; D_i)$ is the count of term t in document D_i and $P(t|C)$ the probability of observing t in the collection. Relevance models perform better when they are subsequently interpolated with the original query using a mixing weight λ [4]:

$$P(t|\theta_Q) \quad = \quad \lambda \cdot \frac{c(t; Q)}{|Q|} + (1 - \lambda) \cdot P(t|\hat{\theta}_Q), \qquad (3)$$

where $|Q|$ denotes the length of the query.

Parsimonious language models may be used to reduce the amount and probability mass of non-specific terms in either queries, documents, or feedback documents by iteratively adjusting the individ-

SIGIR'08, July 20–24, 2008, Singapore.
ACM 978-1-60558-164-4/08/07.

ual term probabilities based on a comparison with a large reference corpus, such as the collection [3]. While relevance models already contain a way of incorporating a reference corpus, viz. Eq. 2, we propose to make the estimate $P(t|\hat{\theta}_q)$ more sparse. Doing so would enable more query-specific terms to receive more probability mass, thus making the resulting query model more to the point. We approach this by parsimonizing the individual estimates $P(t|D)$ in Eq. 1 through applying the following EM algorithm until the estimates do not change significantly anymore:

E-step: $$e_t = c(t;D) \cdot \frac{\gamma P(t|D)}{(1-\gamma)P(t|C) + \gamma P(t|D)},$$

M-step: $$P(t|D) = \frac{e_t}{\sum_{t'} e_{t'}}.$$

3. RESULTS AND DISCUSSION

To measure the effectiveness of our proposed feedback approach, both compared to a baseline and to relevance models, we use test collections from four genres (news, domain-specific, intranet, user generated content), using only the title field:

1. **TREC disks 4 and 5, minus the Congressional Record**, with 50 topics from TREC-7 ad hoc track [10],
2. **TREC disks 4 and 5, minus the Congressional Record**, with 250 topics from TREC Robust 2004 [11],
3. **TREC Blog** with two times 50 topics from 2006 and 2007 [6] track test collections,
4. **TREC Genomics** with 36 topics from 2007 [2], and
5. **TREC Enterprise** with 50 topics from 2007 (document search task); results are reported only for relevance level 1 [8].

Test collection	Run	MAP	P@10	MRR
TREC-7	QL	0.1642	0.3760	**0.6295**
	RM	0.1747	0.3640	0.5618
	PRM	**0.2091**†/‡	**0.4120**†/‡	0.5662‡
TREC Robust 2004	QL	0.2247	0.3968	0.6098
	RM	0.2430†	0.4056	0.6050
	PRM	**0.2689**†/‡	**0.4289**†/‡	**0.6115**†/‡
TREC Blog 2006	QL	0.3213	**0.6720**	**0.7236**
	RM	0.3313	0.6380	0.6983
	PRM	**0.3379**‡	0.6700‡	0.7206
TREC Blog 2007	QL	0.4327	0.6820	0.7558
	RM	0.4371	0.6780	0.6929
	PRM	**0.4571**‡	**0.7280**‡	**0.7629**
TREC Genomics 2007	QL	0.2695	0.4306	0.6098
	RM	0.2828	0.4389	0.5732
	PRM	**0.2850**	**0.4528**	**0.6196**
TREC Enterprise 2007	QL	0.3552	0.7100	0.8583
	RM	0.4227†	0.6940	0.8304
	PRM	**0.4433**†	**0.7400**‡	**0.8597**

Table 1: Results per test collection for the baseline query-likelihood run (QL), relevance models (RM), and parsimonious relevance models (PRM) (best results are marked in boldface). †/‡ indicates a statistically significant difference as compared to the baseline or to the RM run respectively, using a two-tailed paired t-test at $\mathbf{p < 0.01}$.

For each topic set we construct three runs: (i) a baseline query-likelihood run without any relevance feedback or parsimonization (QL) [7], (ii) a run based on blind relevance feedback with Lavrenko's relevance model (RM) [5], and (iii) a run using blind relevance feedback with parsimonized relevance models (PRM). We fix $\gamma = 0.15$ [3] and sweep over possible values for λ and $|\mathcal{D}_Q|$. We report on mean average precision (MAP), precision at 10 (P@10), and mean reciprocal rank (MRR) using the optimal parameter settings (which were obtained empirically).

The results of our experiments are listed in Table 1. The scores of the baseline approach are at the same level as, or better than, the median scores at the corresponding TREC task. From Table 1 we arrive at 3 observations: (i) parsimonizing relevance models has a positive effect on retrieval effectiveness in terms of MAP on all collections; most interesting are the statistically significant improvements on TREC Robust 2004, since this specific collection is known for its difficulty at handling relevance feedback; (ii) parsimonizing relevance models improves the performance of these models on all measures, and in most cases significantly so; (iii) in most test settings the parsimonious relevance models improve retrieval performance with regard to early precision and reciprocal rank, even though blind relevance feedback is considered to only have a recall enhancing effect [9].

4. CONCLUSIONS AND FUTURE WORK

We have used parsimonious language models to re-estimate term probabilities assigned by relevance models. We have evaluated the method on five test collections involving four document genres. Results show that parsimonious relevance models (i) improve retrieval effectiveness in terms of MAP on all collections, (ii) significantly outperform their non-parsimonized counterparts on most measures, and (iii) have a precision enhancing effect, unlike other blind relevance feedback methods.

5. ACKNOWLEDGEMENTS

This work was carried out in the context of the Virtual Laboratory for e-Science project. The work was also supported by by the Netherlands Organisation for Scientific Research (NWO) under project numbers 220-80-001, 017.001.190, 640.001.501, 640.002.-501, STE-07-012 and by the E.U. IST programme of the 6th FP for RTD under project MultiMATCH contract IST-033104. .

6. REFERENCES

[1] P. Anick. Using terminological feedback for web search refinement: a log-based study. In *SIGIR '03*, 2003.
[2] W. Hersh, A. Cohen, and P. Roberts. TREC 2007 Genomics track overview. In *TREC 2007 Working Notes*, 2007.
[3] D. Hiemstra, S. Robertson, and H. Zaragoza. Parsimonious language models for information retrieval. In *SIGIR '04*, 2004.
[4] O. Kurland, L. Lee, and C. Domshlak. Better than the real thing?: iterative pseudo-query processing using cluster-based language models. In *SIGIR '05*, 2005.
[5] V. Lavrenko and W. B. Croft. Relevance based language models. In *SIGIR '01*, 2001.
[6] C. Macdonald and I. Ounis. The TREC Blog 06 collection: Creating and analyzing a blog test collection. University of Glasgow, 2006.
[7] J. M. Ponte and W. B. Croft. A language modeling approach to information retrieval. In *SIGIR '98*, 1998.
[8] I. Soboroff, A. de Vries, and N. Craswell. Overview of the TREC 2007 Enterprise Track. In *TREC 2007 Working Notes*, 2007.
[9] B. Vélez, R. Weiss, M. A. Sheldon, and D. K. Gifford. Fast and effective query refinement. In *SIGIR '97*, 1997.
[10] E. Voorhees. Overview of TREC 2002. In *Proceedings of the 11th Text Retrieval Conference (TREC 2002)*, 2002.
[11] E. M. Voorhees. The TREC Robust retrieval track. *SIGIR Forum*, 39 (1):11–20, 2005.
[12] J. Xu and W. B. Croft. Query expansion using local and global document analysis. In *SIGIR '96*, 1996.
[13] C. Zhai and J. Lafferty. Model-based feedback in the language modeling approach to information retrieval. In *CIKM '01*, 2001.